

Intelligent scaling of resources in edge-cloud systems using the Analytic Hierarchy Process

Marko Marković¹, Petar Spalević², Dragan Soleša³

¹University Business Academy in Novi Sad, Faculty of Applied Management, Economics and Finance, Belgrade, Serbia

²University of Pristina, Faculty of Technical Sciences, Kosovska Mitrovica, Serbia

³University Business Academy in Novi Sad, Faculty of Economics and Engineering Management, Novi Sad, Serbia
E-mail: marko.markovic@mef.edu.rs

Abstract. Higher education institutions are increasingly using cloud computing solutions to enable the smooth operation of their systems, which often face issues with sudden spikes in load, most seen in student portals and learning management systems (LMS). The key challenge in such an environment is selecting an appropriate automatic scaling mechanism that can balance performance and costs. This paper applies the Analytic Hierarchy Process (AHP) method for the multi-criteria evaluation of three scaling mechanisms: reactive, proactive, and hybrid. With the help of IT experts, five key criteria were defined that contribute to better decision-making. The following criteria were used: latency, cost, implementation complexity, operational stability, and processor load. Based on the analysis conducted, the results show that the most significant factor in decision-making is operational stability, with a weight coefficient of 44.1%. The final ranking established that the proactive model is in first place with a value of (0.5722), which, thanks to its ability to predict loads, provides reliability and quality of service for the needs of the higher education system.

1 Introduction

The modern model of cloud computing has significantly influenced the way organizations provide and manage computing resources and their infrastructure over the past few years [1]. In order to transform this process, access to scalable infrastructure is provided as needed by users or organizations [2]. This change in approach can particularly be applied in sectors where a sudden spike in load is expected; in terms of higher education institutions, all services can be migrated to cloud infrastructure, such as student profiles, LMS systems, or examination systems. The advantage lies in the ability of cloud computing technology to dynamically adjust capacity to provide resources in response to sudden spikes in workload, thereby improving performance while reducing costs [3]. One of the key features is elasticity, as it allows for efficient management of unpredictable spikes in load that are typical for web services of higher education institutions. However, unlike commercial solutions, these services have specific patterns of load. Student services often experience sudden and unpredictable spikes, especially during exam registrations or assignment submissions, unlike commercial solutions that have more stable user flows. Sudden spikes can burden the technical infrastructure, which in turn leads to slow response times, outages, and poor user experience. To eliminate potential risk, it is necessary to use reactive scaling techniques that are essential for such situations. But a problem with execution delays may arise here, which could disrupt the operation of the system [4]. Additionally, industrial tools

such as AWS Auto Scaling or Kubernetes HPA most commonly use set threshold values, typically CPU utilization, as a trigger to initiate mechanisms, which can lead to delays or inefficient resource allocation during load [5]. The outcome is often a problem with latency or unnecessary resource consumption. Automatic scaling encompasses three categories of mechanisms: reactive, proactive, and hybrid approaches [6]. The reactive approach is based on monitoring the system's behavior in real time, based on CPU utilization, memory consumption, and latency. It activates its mechanisms at the moment when a defined threshold is exceeded [7]. The reactive model is very simple to implement because it utilizes the delay time between the occurrence of a sudden spike in load and the completion of the system's response in its operation. Such an approach leads to a deterioration in system performance and temporary interruptions [8]. Proactive or predictive automatic scaling, as the term itself indicates, utilizes data and models of past behavior that include time series analysis, deep learning techniques, and machine learning algorithms to forecast future resource allocation needs and to proactively initiate certain actions even before a change in load occurs [9]. This model has demonstrated capabilities in reducing latency and improving resource utilization, for example, mechanisms that use machine learning significantly reduce latency and achieve better resource allocation compared to a reactive approach [10]. The problem arises due to inaccurate predictions under conditions of new constant load, as they introduce additional complexity in terms of model training and parameter tuning [11]. The hybrid mechanism combines all the advantages offered by reactive and proactive model strategies, using proactive scaling for prediction while retaining the reactive mechanism as protection against unexpected oscillations [12]. In addition to integrating multiple scaling methods, the implementation itself within the infrastructure presents numerous challenges due to increased configuration complexity and the burden on computing resources. Complexity can also manifest as instability caused by excessively frequent scaling actions, which can lead to oscillations in resource allocation. A major problem in decision-making is the choice of the appropriate automatic scaling mechanism, as it requires consideration of multiple conflicting criteria, the most prominent of which are financial costs, latency, implementation complexity, operational stability, and resource utilization efficiency [13]. Automatic scaling strategies claim to be efficient, but they must consider not only the technical characteristics of behavior but also the financial aspect of the business, as different cloud

platforms apply specific billing techniques (from on-demand instances to spot markets and discounts for long-term use). This problem poses a significant challenge for the IT department, which has the task of ensuring the constant operation of the system while minimizing resource and financial consumption. The choice between reactive, predictive, and hybrid automatic scaling represents a multi-criteria decision-making (MCDM) problem. Automatic scaling can be reactive, proactive, or hybrid. The reactive approach responds after the emergence of load, the proactive approach anticipates future resource needs, while the hybrid approach combines both methods for greater robustness. The comparison of key indicators is shown in the table below.

Table 1. Quantitative Comparison of Autoscaling Approaches

Model	Latency	Resources	Reaction	SLA
Reactive [6]	High	60-70%	Slow	23%
Proactive [8]	↓95%	< 90%	Quick	Low
Hybrid [12]	↓20-30%	80-90%	Quick	6%

2 Research Methodology

2.1 AHP Method

In this work, the method of multi-criteria decision-making is employed using the Analytic Hierarchy Process (AHP), which was developed in 1980. in the year by Thomas Saaty. AHP enables the evaluation of alternatives by structuring complex problems into a hierarchical framework that encompasses criteria, alternatives, and objectives as a synthesis of quantitative metrics and qualitative assessments using the pairwise comparison method [14]. This method is applicable in various domains and can be used in the context of cloud computing for making decisions about resource management and scaling. After the structure is built, the respondents will define their opinions for each of those levels. The comparison is made using a nine-point numerical scale (known as the fundamental scale of comparison), proposed by Saaty [15]. Table 2 presents the scale with an explanation of each value.

Table 2. Fundamental Scale of Comparison

Intensity	Definition	Explanation
1	Equal importance	Both activities are equal.
3	Moderate Importance	One activity is slightly favored.
5	Strong Importance	One activity is strongly favored.
7	Very Strong Importance	One activity is very strongly favored.
9	Extreme Importance	One activity is extremely favored.
2, 4, 6, 8	This is a compromise	The decision maker is undecided.

The comparison will be carried out in pairs using a square comparison matrix ($n \times n$) $A = [a_{ij}]$, assuming that the value of the estimate for the less important element is the inverse of the estimate of the value of the more important element according to the decision-maker's opinion, i.e. for the matrix, for which (1) holds [16]:

$$a_{ij} = \frac{1}{a_{ji}} \text{ and } a_{ij} = 1, \text{ for each } i, j, k = 1 \dots n \quad (1)$$

Then, for each matrix of even comparisons, we need to calculate the priority vector according to the following formula (2):

$$A_w = \lambda_{max} w \text{ where } \lambda_{max} \text{ is the principal eigenvalue of } A \quad (2)$$

Then, for the given values, the consistency index of the matrix CI (3) is calculated:

$$CI = \frac{\lambda_{max} - n}{n - 1} \text{ where } n \text{ is the number of criteria} \quad (3)$$

Next, we calculate the relative consistency coefficient CR (4):

$$CR = \frac{CI}{R}, \text{ where each } R \text{ maps to a matrix element} \quad (4)$$

According to Sathy, the value of R is 0 to order 2, while for numbers from 3 to 10 the index values are of random consistency, namely: 0.58, 0.90, 1.12, 1.24, 1.32, 1.41, 1.45 and 1.49. A consistency ratio of up to 10% is considered acceptable [16].

2.2 The data used in the research

One of the most important advantages of this approach is the ability to incorporate expert knowledge into a mathematically consistent framework. Given that the assessment of automatic scaling mechanisms for higher education institutions often involves criteria that may not be directly comparable, with the most important being quantitative criteria such as latency and CPU usage, and qualitative criteria such as implementation complexity and operational stability, the multi-criteria decision-making method can be the most important tool in creating a framework for integrating such heterogeneous evaluation criteria. Within this research, structured pairwise comparative assessments were obtained from four IT experts with direct operational experience in managing the infrastructure of a higher education institution. The main goal of this research is to rank three mechanisms of automatic scaling: reactive, predictive, and hybrid, using the AHP method. The ranking is carried out using five criteria that together encompass the main dimensions and fundamentals of automatic scaling: Latency represents the ability of the mechanism to provide an acceptable response time of the service during a spike in load. This criterion is often encountered as key in the evaluation of cloud auto-scaling [17]. The cost reflects the need to secure resources in every scaling strategy, including the costs of the entire infrastructure; sometimes, other costs may arise due to the scaling operations themselves [18]. The complexity of implementation relates to the level of expertise and the complexity of the infrastructure required for implementation, as well as the maintenance of each approach to automatic scaling. It is most characterized as a barrier, as some teams lack the expertise to apply advanced scaling techniques [19]. Operational stability is the degree to which a mechanism avoids oscillatory behavior and ensures adequate resource allocation, along with precise scaling estimates that can lead to instability and a reduction in service reliability [19]. The CPU load refers to the efficient use of computing resources, as well

as avoiding excessive or insufficient resource reservation [13]. The contributions of this work can be viewed in two segments. The first is that this work helps in solving the problem of choosing an automatic scaling strategy, given that this topic is still insufficiently studied in terms of existing literature. The second contribution of the work is the use of the analytic hierarchy process as a decision-making tool, as this approach allows for addressing the technical difficulties that arise during automatic scaling in the cloud.

3 Results

Given that four experts participated in the research, it is necessary to combine the opinions and attitudes of all four respondents using the geometric mean, using the following formula (5):

$$GM_{ij} = \sqrt[4]{E_1 \times E_2 \times E_3 \times E_4} \quad (5)$$

Table 3 shows the initial matrix of even comparison for the five defined criteria. The values in the cells represent a group estimate of four IT professionals, obtained using the geometric mean method.

Table 3. Determining the weight of the criteria

Criteria	C_1	C_2	C_3	C_4	C_5	Weight
C_1 Latency	1	3	5	1/2	2	0.2596
C_2 Cost	1/3	1	2	1/5	1/2	0.0882
C_3 Impl. Complexity	1/5	1/2	1	1/7	1/4	0.0497
C_4 Operational Stability	2	5	7	1	3	0.4406
C_5 Processor Load	1/2	2	4	1/3	1	0.1619

Based on the pairwise comparison matrix of the criteria provided in Table 3, the consistency calculation was carried out using formulas (3) and (4). The maximum calculated eigenvalue is $\lambda_{max} = 5.0416$, which provides a consistency index of $CI = 0.0104$. Using the random index value for $n = 5$ ($RI = 1.12$), the final consistency ratio was calculated as $CR = 0.0092$. From the presented results, it can be concluded that the criterion of operational stability achieved the highest weight coefficient with a value of 0.4406 (44.1%). This means that reliable operation without fluctuations is very important in the cloud environment of higher education institutions. After that, the second in line is latency with a weight of 0.2596 and CPU load at 0.1619, which confirms that performance and response speed are crucial for service quality. On the other hand, the criteria related to costs and implementation complexity have a minor impact, which suggests that institutions are willing to invest in infrastructure if it can ensure quality and uninterrupted operation, so implementing a more expensive and complex technical solution will not be a problem.

Table 4. Prioritization of criteria for three alternatives

Criteria	Reactive	Proactive	Hybrid
C_1 Latency	0.1429	0.5714	0.2857
C_2 Cost	0.6687	0.0882	0.2431
C_3 Impl. Complexity	0.7132	0.0669	0.2200
C_4 Operational Stability	0.0978	0.7151	0.1871
C_5 Processor Load	0.1851	0.6031	0.2118

Table 4 shows the priorities of the criteria for three scaling mechanisms, according to five considered criteria. The values in the table show how much each scaling approach amounts to within a specific criterion, and the global weight coefficients calculated during the previous phase of analysis are indicated in parentheses next to the criteria. It is important to note that the proactive approach shows the best results according to some very important criteria such as operational stability and latency. While the reactive model has shown good results in terms of cost and low implementation complexity.

Table 5. Overall Global Priorities of Alternatives Based on the AHP Method

Criteria	Reactive	Proactive	Hybrid
C_1 Latency	0.0371	0.1483	0.0742
C_2 Cost	0.0590	0.0078	0.0214
C_3 Impl. Complexity	0.0354	0.0033	0.0109
C_4 Operational Stability	0.0431	0.3151	0.0824
C_5 Processor Load	0.0300	0.0977	0.0343
	0.2046	0.5722	0.2233

Based on the analysis conducted and the results presented in Table 5, the obtained values define the optimality of each of the three considered mechanisms. In first place is the proactive model with a total global priority of 0.5722, which clearly demonstrates its dominance in the performance domain of an adequate mechanism for complex Cloud environments. In the individual results, the criteria that recorded slightly lower scores are cost and ease of implementation, but its greatest advantage is expressed in two critical factors: latency and operational stability. Given that these two criteria together account for over 70% of the weighted impact on the final decision, the ability of the proactive algorithm to predict spikes in load has proven to be crucial, rendering infrastructural complexity negligible in comparison to system reliability. The hybrid model achieved a score of 0.2233, thus securing second place. In the hybrid model, a consistent approach stands out, but when sudden spikes in load occur, it does not provide the same level of prevention as the proactive model. In last place is the reactive model with a score of 0.2046, and while it is favorable in terms of cost and ease of implementation, it has been unequivocally proven that speed and quality of service are crucial for higher education institutions, and that the economic aspect loses significance compared to the stable operation of the system.

4 Discussion

To ensure smooth resource provisioning to all users, it is necessary to design and implement an architecture capable of responding to dynamic workload changes. The starting point of this research was the selection of an optimal scaling mechanism, which represents a multi-criteria decision-making problem due to the presence of several conflicting criteria. The results show that operational stability is the most significant criterion, with the highest weight coefficient of 0.4406. This confirms that higher education institutions value not only fast system response, but also the reliability and continuity of

service operation [20]. Even minor fluctuations in resource allocation can negatively affect user experience, especially during sudden workload increases. Latency and processor load also have an important role, with weights of 0.2596 and 0.1619, while cost and implementation complexity have lower influence on the final decision. When observing the overall results of the alternatives, the proactive model achieved the highest global priority value of 0.5722. Its advantage is reflected in the ability to predict workload increases and react before the system reaches a critical state. This directly contributes to lower latency and higher operational stability. Although this model has lower performance in terms of cost and implementation complexity, these limitations do not significantly affect its final ranking because the most important criteria are related to system reliability and performance. The hybrid model achieved a global priority value of 0.2233, which places it in second position. This model represents a compromise between reactive and proactive approaches, but the obtained results show that combining different mechanisms does not necessarily lead to optimal performance. Although it provides balanced results, it does not reach the level of efficiency achieved by the proactive model. The reactive model achieved the lowest global priority value of 0.2046. Although it is the simplest and most cost-effective solution, its main limitation is delayed response after workload increase has already occurred. Recent monitoring-driven and workload-adaptive approaches further demonstrate the importance of reducing dependence on fixed, resource-specific thresholds. Work presented at the MODA and EESP workshops at ISC High Performance 2026 emphasizes the use of operational telemetry, workload characterization, and adaptive scheduling to improve resource-management decisions across heterogeneous computing environments [21,22,23]. Such approaches represent a complementary direction for addressing the delayed response and limited adaptability associated with conventional reactive scaling mechanisms. In dynamic environments, this can lead to performance degradation, increased latency, and reduced service stability. Therefore, the results indicate that the proactive scaling mechanism is the most suitable solution for edge-cloud systems in higher education institutions.

5 Conclusion

The research conducted in this paper dealt with the selection of an adequate mechanism for automatic resource scaling in edge-cloud systems. Starting from the basic facts that sudden load spikes occur in higher education institution systems, the established model allows for the analysis and evaluation of alternative scaling approaches using the AHP multi-criteria decision-making method. The results show that the key factor is operational stability with the highest weight coefficient, which clearly emphasizes the importance of system operation without oscillations in relation to other criteria. Latency and resource utilization efficiency also have a significant impact, while costs and implementation complexity are somewhat less recognized as factors. By analyzing the alternatives, the proactive automatic scaling model proved to be the most reliable solution compared to other alternatives. The ability to predict future loads and the possibility of timely reaction significantly reduces

latency and avoids bringing the system into a critical state. Given that it implies greater implementation complexity and higher implementation costs, the results show that these two shortcomings do not play a major role in the final selection decision. The hybrid model is in the middle, both in the segment that combines reactive and proactive approaches, and in the source itself among the alternatives. Unfortunately, it does not reach a high level of efficiency, and for this reason it is considered a little weaker than the proactive model. The reactive model, in addition to being simple and economical, has problems in dynamic environments, due to its delay in moments of dynamic load, which leads to a decrease in performance and quality of service. The contribution of this work is reflected in the development of a well-founded decision-making framework, which integrates expert knowledge and allows for the comparison of criteria and alternatives in an adequate manner. The work also has practical value, as it can serve as a basis for implementing and optimizing cloud infrastructure in systems that have a similar load behavior model. Future research can extend the given model by integrating advanced machine learning techniques into hybrid models, all with the aim of hybrid models achieving the best possible balance between system performance, operational stability, and costs.

Literature

- [1] A. Veljić, M. Marković, and S. Maričić, "Innovations in the architecture of distributed computing systems," in *Proc. Int. Scientific & Professional Conf. Innovation as an Initiator of the Development 2025*, pp. 401-407, 2025, ISBN: 978-86-84531-82-9.
- [2] A. Ilyushkin *et al.*, "An Experimental Performance Evaluation of Autoscalers for Complex Workflows," *ACM Trans. Model. Perform. Eval. Comput. Syst.*, vol. 3, no. 2, pp. 1-32, 2018, doi: 10.1145/3164537.
- [3] N. Dang-Quang and M. Yoo, "Deep Learning-Based Autoscaling Using Bidirectional Long Short-Term Memory for Kubernetes," *Applied Sciences*, vol. 11, no. 9, p. 3835, 2021, doi: 10.3390/app11093835.
- [4] E. Park, K. Baek, E. Cho, and I. Ko, "Fully Decentralized Horizontal Autoscaling for Burst of Load in Fog Computing," *Journal of Web Engineering*, 2023, doi: 10.13052/jwe1540-9589.2261.
- [5] H. Lee and M. Kim, "Adaptive, ML-enhanced resource management for high-performance cloud-based web platforms," *Research Square*, preprint, 2025, doi: 10.21203/rs.3.rs-6025433/v1.
- [6] S. Gupta, "Proactive and Reactive Autoscaling Techniques for Edge Computing," *arXiv preprint*, 2025, doi: 10.48550/arXiv.2510.10166.
- [7] V. Podolskiy, A. Jindal, and M. Gerndt, "Multilayered Autoscaling Performance Evaluation: Can Virtual Machines and Containers Co-Scale?," *Int. J. Appl. Math. Comput. Sci.*, vol. 29, no. 2, pp. 227-244, 2019, doi: 10.2478/amcs-2019-0017.
- [8] P. Vemasani, S. Vuppapapati, S. Modi, and S. Ponnusamy, "Achieving Agility through Auto-Scaling: Strategies for Dynamic Resource Allocation in Cloud Computing," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 12, no. 4, pp. 3169-3177, 2024, doi: 10.22214/ijraset.2024.60566.
- [9] B. Predic *et al.*, "Cloud-load forecasting via decomposition-aided attention recurrent neural network tuned by modified particle swarm optimization," *Complex & Intelligent Systems*, 2023, doi: 10.1007/s40747-023-01265-3.
- [10] V. Maksimović, M. Petrović, D. Savić, B. Jakšić, and P. Spalević, "New approach for estimating edge detection

- threshold and application of adaptive detector depending on image complexity,” *Optik - International Journal for Light and Electron Optics*, vol. 238, p. 166476, 2021, doi: 10.1016/j.ijleo.2021.166476.
- [11] O. Pozdniakova, D. Mažeika, and A. Cholomskis, “SLA-Adaptive Threshold Adjustment for a Kubernetes Horizontal Pod Autoscaler,” *Electronics*, vol. 13, no. 7, p. 1242, 2024, doi: 10.3390/electronics13071242.
 - [12] A. Bento, F. Araújo, and R. Barbosa, “Cost-Availability Aware Scaling: Towards Optimal Scaling of Cloud Services,” *Journal of Grid Computing*, vol. 21, no. 4, 2023, doi: 10.1007/s10723-023-09718-2.
 - [13] A. Uta, S. Au, A. Ilyushkin, and A. Iosup, “Elasticity in Graph Analytics? A Benchmarking Framework for Elastic Graph Processing,” in *Proc. IEEE Int. Conf. Cluster Computing (CLUSTER)*, 2018, pp. 381–391, doi: 10.1109/CLUSTER.2018.00056.
 - [14] I. Dimolitsas, D. Spatharakis, D. Dechouniotis, and S. Papavassiliou, “AHP4HPA: An AHP-based Autoscaling Framework for Kubernetes Clusters at the Network Edge,” in *Proc. IEEE GLOBECOM*, 2022, pp. 2566–2571, doi: 10.1109/GLOBECOM48099.2022.10001214.
 - [15] T. L. Saaty, “The analytic hierarchy process-what it is and how it is used,” *Mathematical Modelling*, vol. 9, no. 3-5, pp. 161–176, 1987, doi: 10.1016/0270-0255(87)90473-8.
 - [16] I. I. Karayalcin, “The analytic hierarchy process: Planning, priority setting, resource allocation: Thomas L. Saaty McGraw-Hill, New York, 1980,” *European Journal of Operational Research*, vol. 9, pp. 97–98, 1982, doi: 10.1016/0377-2217(82)90022-4.
 - [17] R. Chowdhury, C. Talhi, H. Ould-Slimane, and A. Mourad, “Proactive and Intelligent Monitoring and Orchestration of Cloud-Native IP Multimedia Subsystem,” *IEEE Open Journal of the Communications Society*, vol. 5, pp. 139–155, 2024, doi: 10.1109/OJCOMS.2023.3341002.
 - [18] J. R. Gunasekaran, “Minimizing cost and maximizing performance for cloud platforms,” in *Proc. 21st Int. Middleware Conf. Doctoral Symp.*, 2020, pp. 29–34, doi: 10.1145/3429351.3431747.
 - [19] M. Dashtbani, “Key Considerations for Auto-Scaling: Lessons from Benchmark Microservices,” *arXiv preprint*, 2025, doi: 10.48550/arXiv.2510.02585.
 - [20] M. Marković and D. Soleša, “Security and Data Protection in Artificial Intelligence,” *Journal of Process Management and New Technologies*, vol. 13, no. 1-2, pp. 113–123, 2025, doi: 10.5937/jpmnt13-58872.
 - [21] B. Ma, J. Eitzinger, and G. Wellein, “Automatic Workload Characterization on Production HPC Systems via Roofline Telemetry,” in *Proc. 7th ISC HPC Int. Workshop on Monitoring, Observability, and Operational Data Analytics (MODA26), ISC High Performance 2026*, Hamburg, Germany, 2026.
 - [22] D. Arshad and M. Weiland, “A study on improving HPC system efficiency through moldable and mobile job scheduling,” in *Proc. 2nd Int. Workshop on Energy Efficiency with Sustainable Performance (EESP), ISC High Performance 2026*, Hamburg, Germany, 2026.
 - [23] V. Nalbalwar, P. Pagar, R. Rokade, S. Patil, and A. Joshi, “Thermal-Aware Core-Level Scheduling for Kubernetes Workloads,” in *Proc. 2nd Int. Workshop on Energy Efficiency with Sustainable Performance (EESP), ISC High Performance 2026*, Hamburg, Germany, 2026.